Received: 26 December 2006 ■ *Revised: 9 November 2007* ■ *Accepted: 14 November 2007* ■ *Publication: 13 February 2008*

JNIRS JOURNAL OF NEAR INFRARED SPECTROSCOPY

SAS® partial least squares for discriminant analysis

James B. Reeves, III[a] and Stephen R. Delwiche[b]

[a]USDA, ARS, Environmental Management and Byproduct Utilization Laboratory, Bldg 306, BARC East, Beltsville, MD 20705, USA.
E-mail: james.reeves@ars.usda.gov

[b]USDA, ARS, Food Safety Laboratory, Bldg 303, BARC East, Beltsville, MD 20705, USA

The objective of this work was to implement discriminant analysis using SAS® partial least squares (PLS) regression for analysis of spectral data. This was done in combination with previous efforts, which implemented data pre-treatments including scatter correction, derivatives, mean centring and variance scaling for spectral analysis. Partial least squares analysis is implemented in SAS® as type 2 where a solution for multiple analytes (*Y*-variables) is determined simultaneously, but cannot work with non-numeric analyte values. For discriminant analysis, samples belonging to one of *Z* classes are coded for *Z* analytes with all but one (class to which sample belongs coded as 1) coded as being a 0. Thus, for four classes, all samples are coded with one of four analyte combinations (1,0,0,0; 0,1,0,0; 0,0,1,0; or 0,0,0,1). This paper discusses a SAS® program designed to perform classification/discriminant analysis using SAS® PLS, and to a smaller extent, principal component analysis and reduced rank regression. The authors' previously written SAS® macros for pre-treatment of spectral data are implemented. Examples are presented using two datasets: forages and by-products, and grains. The program allows for testing of multiple spectral pre-treatments in a one-step fashion with summary of all results. The macro coding for the program and test data sets is available at: http://www.impublications.com/nir/page/software. *Please note that the program will not work properly on Unix-based systems due to DOS calls.*

Keywords: PLS, partial least squares, principal components, PCA, SAS®, discriminant analysis

Introduction

Classification or discriminant analysis using spectral data is commonly performed using various implementations of principal component analysis (PCA) including procedures as simple as plotting one factor against another. More sophisticated methods such as soft independent modelling of class analogy (SIMCA)[1] form a separate model for each class of sample by finding the commonality within those samples. Still other methods such as stepwise discriminant analysis using multi-linear regression are based on finding specific differences between classes of samples. There are many ways to perform discriminant analysis and the objective of this work is not to compare them or discuss their merits, but to implement partial least squares (PLS) as a tool for discriminant analysis as discussed by Barker and Rayens[2] using SAS® PLS.

To use the SAS® PLS procedure for discriminant analysis a method for encoding the different classes is needed. As stated in our previous papers on SAS® PLS,[3,4] while SAS® PLS has a class variable for non-numeric variables, this variable cannot be used as an analyte or *Y*-variable. If only two or, assuming an inherent natural ordering, three classes exist, classes can be coded as –1 and +1, or –1, 0 and +1 (assumes an ordering and equal spacing), and PLS analysis performed as usual in order to carry out discriminant analysis. However, such regressions become inappropriate when several disjoint classes are considered. As outlined in a SPSS (SPSS Inc., Chicago, IL, USA) white paper by J.J. Meulman,[5] multi-class PLS classifi-

The authors wish to state the use or mention of any commercial software does not imply any endorsement of that product or any better suitability than other similar products by either the authors, the US Department of Agriculture or IM Publications.

ISSN: 0967-0335
doi: 10.1255/jnirs.757

cation is implemented through the creation of one *Y*-variable for each class and to code a sample as a 1 for the class to which it belongs and as 0 for all other classes (for example, 0,1,0,0, for a sample belonging to the second of four classes). One then uses PLS2 and finds a solution for all *Y*-variables (classes) simultaneously. As outlined in our first paper on SAS® PLS,[3] PLS2 is the method implemented in SAS®, which was an impediment for quantitative analysis, but is now an asset for discriminant analysis.

Spectral pre-treatments, such as multiplicative scatter correction (MSC)[6] or standard normal variate (SNV) transformation,[7] can greatly improve spectral calibrations for quantitative analysis by removing additive and multiplicative artifacts. Other pre-treatments, such as derivatives,[8] can accentuate the differences between spectra in regions of component-related absorption. Because discriminant analysis using PLS is based on finding differences between classes based on spectral differences, these pre-treatments may also be useful for discriminant analysis. The objective of this work was to implement a SAS® based PLS program for discriminant analysis which included data pre-treatments such as derivatives and scatter correction that allows the analyst to select the optimal pre-treatment in a single (batch) submission of the program.

Experimental

Samples

Two different historical datasets were utilised in the testing of the program presented. Set 1 consisted of 241 forage samples composed of 174 chlorite treated samples (16 different forages and by-products) and 67 samples (five forages, many same species as in the treated samples) collected at different stages of growth in a single growing season. Samples were initially coded as one of 14 classes as several forages were present in both sets (for example, alfalfa hay both treated and at different stage of growth or material from more than one source, two sources of corn and soybean stover). Further information on these samples made be found in earlier publications.[9,10] Set two consisted of 3600 single-seed spectra of durum (tetraploid) wheat from 75 breeders' lines of 48 seeds (=48 spectra) per line. The lines were categorised by one of four genotypic conditions of the waxiness gene that encodes for the production of the enzyme, granule bound starch synthase (GBSS), which regulates amylose production. Full details are provided in a recent publication.[11]

Basic SAS® program structure

The SAS® program consists of sections of modular code called macros. Many of these macros [for the following tasks: reading a file containing the analyte values; preparing analyte and spectral values for processing; performing gap and Savitzky–Golay derivatives;[8] averaging or skipping spectral data points; performing multiplicative scatter correction (MSC)[6] or standard normal variate (SNV) transformation,[7] with or without detrend; mean centring and variance scaling; and preparing all data for PLS processing] are the same as discussed in our earlier publications on SAS® PLS for quantitative analysis[3,4] and are briefly identified in the appendix. Code for these previous programs and the newer discriminant version is available at the website: http://www.impublications.com/nir/page/software. It should be noted that due to DOS calls, as presently written the PLS part of the program will not operate on Unix-based systems. Recompiling and linking of "X" command calls and DOS programs would be required.

Results and discussion

As previously stated, the objective of this work was to present an SAS® implementation for discriminant analysis using PLS, rather than to compare discriminant analysis algorithms. As such, discussion will be limited to the format of the output produced and a few other pertinent findings. Results for a typical analyte are shown in Tables 1–3. As shown in Table 3, the first output consists of a listing of the parameter settings used for the run in question. This is very useful in checking to be sure that the settings used were as intended. These settings are set specifically in the SAS® code as outlined in the program documentation prior to running the program.

The results for one analyte of a typical run are shown in Table 1. The printed results are based on factor selection according to SAS® PLS[12] as previously discussed. This method of determining the number of factors was previously found to often select a greater number of factors, typically by one or two, than the more conservative F-Test.[13]

While the output typically contains information for both a calibration set and test set, there are two occasions when this may not be true. If the parameter "NUMCALFILES" is set to "NUMFILES" (see Table 3), then no validation set or test set is created. Also, if the validation/test set is selected randomly by setting the "RANDOMIZE" parameter to "YES", then it is possible with small data sets for some of the analytes not to be represented in the validation/test set. In using the forage test set, the split was always ⅔ for calibration and ⅓ for validation and in several runs some classes were not represented in the validation set. While this never happened for the example calibration set, the same problem could occur in calibrations as the number of classes increases with respect to the set's number of samples. Finally, it is possible to split the input data set into calibration and validation sets by putting all the calibration samples at the beginning of the file followed by the validation samples. Under these circumstances, one can ensure that all classes are represented in both calibration and validation sets by setting the parameter "RANDOMIZE" to "NO" at program line 67.

The forage data set consisted of 241 files containing 700 data points and a maximum of 14 classes (forages and by-products), which is a relatively small file. In order to test the effect of more samples, the grain test set was used. This file consisted of 3600 single-seed spectra, but only 128 data points per spectrum and just four classes (wild type, waxy

Table 1. Typical output from DISCRIM program based on factor determination by SAS®.[3,4,12]

RESULTS OF PLS THE SAS WAY—MC + VS BY PROGRAM ONLY[a]							
11:50 Tuesday, September 26, 2006							
Obs	IDERIVAT	IFORM	IMCVS	IGAPS	CHEMANALYTES1	OANALYTE1	FACTOR
1	2ND	STR	SAS	64	0.67685	0.62987	15
2	NONE	MSC	SAS	0	0.68204	0.68241	15
3	1ST	MSC	SAS	64	0.68446	0.64615	15
4	1ST	MSC	SAS	8	0.70183	0.70657	15
5	NONE	STR	SAS	0	0.71927	0.68251	15
6	1ST	STR	SAS	8	0.72113	0.68573	15
7	1ST	STR	SAS	16	0.72122	0.66648	15
8	1ST	MSC	SAS	32	0.72443	0.66855	15
9	2ND	STR	SAS	16	0.72463	0.63377	15
10	1ST	STR	SAS	4	0.72523	0.69019	15
11	1ST	MSC	SAS	4	0.72790	0.67677	15
12	1ST	MSC	SAS	16	0.72969	0.70084	15
13	2ND	STR	SAS	32	0.73440	0.71101	15
14	2ND	MSC	SAS	32	0.73942	0.66512	15
15	1ST	STR	SAS	32	0.74155	0.69303	15
16	2ND	MSC	SAS	16	0.74793	0.68784	15
17	2ND	MSC	SAS	64	0.75095	0.73041	15
18	1ST	STR	SAS	64	0.75812	0.65854	15
19	2ND	MSC	SAS	4	0.77241	0.73788	15
20	2ND	STR	SAS	4	0.77881	0.69921	15
21	2ND	STR	SAS	8	0.78116	0.71327	15
22	2ND	MSC	SAS	8	0.78451	0.77208	15

[a]Obs = Specific spectral pre-treatment tested; IDERIVAT = 1st , 2nd, or no gap derivative; IFORM = Type of scatter correction applied, STR = none, MSC = Multiplicative scatter correction; IMCVS = Mean centring and variance scaling, SAS = By SAS itself; IGAPS = Derivative gap size; CHEMANALYTES1 = R^2 for calibration results, 1 stands for analyte number tested; OANALYTE1 = R^2 for validation or test set; FACTOR = Number of factors used in calibration.

and two separate classes of partial waxy) to discriminate. For many algorithms, SAS® provides information on how the time for execution scales with sample number etc., but this is not true for PLS. Using the entire grain set, the DISCRIM program ran for nearly 300 hours to complete the results for 22 different spectral pre-treatments on a 2.52 GHz Pentium® 4 with 2 GB of RAM using SAS® version 9.1. (The results of the PLS discriminant analysis indicated that this method was on a par with our earlier published results,[11] which used linear discriminant analysis on PCA scores to correctly classified waxy category with >95% accuracy, but produced much lower accuracies for the three other categories.) Discussion with SAS® support indicated that the use of multi-core CPUs or a 32 bit version of SAS® would not be likely to result in any significant increases in computational speed. Based on past experiences with SAS® PLS, it appears to scale poorly with increasing numbers of samples, especially when the number approaches 1000. By reducing the number of spectral pre-treatments tested, or by using fewer samples during initial testing, the time to process the data can be reduced to hours or days rather than weeks.

Executing the program two or more times with a random selection of samples for the calibration and validation sets allows one to see how sample distributions can affect calibrations and can be useful in finding a robust calibration. The results shown in Table 2 are a summary of all calibrations carried out using the forage samples with classes (variable ANALYTEID) based on botanical origin. These summaries are produced by a separate macro procedure ("DISCRIMSUMMARY"), not present in the original program.[3] Summaries by math pre-treatment are also produced (data not shown). When applied to the forage samples, discriminant analysis based on PLS seemed to slightly outperform those based on PCR.

Plots of predicted versus actual results, as demonstrated in Figure 1, can also be output but, unless the variance of the predictions among classes is equal, can be misleading and should not be relied upon as the only measure of classification accuracy; also, reliance on this type of comparison analysis, when the number of classes increases beyond two, becomes problematic.

Formation of the various classes is usually based on prior knowledge of the chemical, physical or structural property of interest. However, numerous unsupervised learning algorithms exist to categorise samples, based strictly on spectral response. This approach may be useful in consolidating the total number of classes to a smaller number. Using the

Table 2. Typical summary of statistical results from 25 runs using random selections of samples for calibration and validations sets combining all pre-treatments tested, 22 × 25 runs = 550. ANALYTEID = Which class tested, METHOD: PLS = partial least squares regression, PCR = principal components and RRR = Reduced Rank Regression, _TYPE_ = Dummy variable produced by SAS, _FREQ_ = Total number of calibrations tested, MNNUMBEROFFACTORS = Mean number of factors, MNRMSD = mean RMSD (relative mean squared deviation for calibration set, MNVRMSD = MNRMSD for validation set, MNCALR2, Mean calibration rsquare, MNTESTR2 = Mean validation or test set rsquare. (First column is observation number).

	ANALYTEID	METHOD	_TYPE_	_FREQ_	MNNUMBER OFFACTORS	MNRMSD	MNVRMSD	MNCALR2	MNTESTR2
1	1	PCR	0	550	13.3618	0.23057	0.26514	0.65711	0.58516
2	1	PLS	0	550	13.8600	0.20486	0.24290	0.72876	0.65210
3	1	RRR	0	550	1.2582	0.25706	3.94621	0.53492	0.09097
4	2	PCR	0	550	13.3618	0.22645	0.24349	0.38063	0.30079
5	2	PLS	0	550	13.8600	0.20215	0.22878	0.50292	0.38994
6	2	RRR	0	550	1.2582	0.28013	0.44859	0.03993	0.01705
7	3	PCR	0	550	13.3618	0.11042	0.12262	0.71463	0.62117
8	3	PLS	0	550	13.8600	0.09516	0.10909	0.78801	0.69078
9	3	RRR	0	550	1.2582	0.20650	0.26553	0.03575	0.02957
10	4	PCR	0	550	13.3618	0.25826	0.28473	0.41179	0.32773
11	4	PLS	0	550	13.8600	0.23038	0.26990	0.53016	0.39853
12	4	RRR	0	550	1.2582	0.32517	0.65414	0.04828	0.02378
13	5	PCR	0	550	13.3618	0.30822	0.34214	0.41045	0.29956
14	5	PLS	0	550	13.8600	0.26352	0.30653	0.56613	0.43397
15	5	RRR	0	550	1.2582	0.23148	3.87876	0.62138	0.09031
16	6	PCR	0	550	13.3618	0.17616	0.19950	0.65475	0.58604
17	6	PLS	0	550	13.8600	0.15749	0.18441	0.72000	0.64293
18	6	RRR	0	550	1.2582	0.29223	0.45983	0.04450	0.02984
19	7	PCR	0	550	13.3618	0.10546	0.11915	0.70671	0.63320
20	7	PLS	0	550	13.8600	0.08766	0.10282	0.79671	0.71624
21	7	RRR	0	550	1.2582	0.19494	0.23903	0.03160	0.03585
22	8	PCR	0	550	13.3618	0.05462	0.06144	0.92693	0.88655
23	8	PLS	0	550	13.8600	0.04638	0.05348	0.94882	0.91348
24	8	RRR	0	550	1.2582	0.20262	0.26134	0.05339	0.04627
25	9	PCR	0	550	13.3618	0.05438	0.06048	0.92631	0.88222
26	9	PLS	0	550	13.8600	0.05258	0.05894	0.93096	0.88692
27	9	RRR	0	550	1.2582	0.20732	0.30047	0.06176	0.05278
28	10	PCR	0	550	13.3618	0.14816	0.16237	0.72597	0.68072
29	10	PLS	0	550	13.8600	0.13204	0.15030	0.78261	0.72642
30	10	RRR	0	550	1.2582	0.27670	0.41797	0.04293	0.03352

forage sample calibration set ($n = 161$ samples in 14 initial classes as an example, the principal component scores from factors 1 to 10 (pretreated with a 11-point second derivative Savitzky–Golay quadratic function convolution) were used to determine the distances between samples, thereby collapsing one or more neighbouring classes into a single class. The dendrogram in Figure 2 indicates that four of the 14 forage classes (PH, CC, RH and SH = peanut hulls, corn cobs, rice hulls and soybean hulls, respectively) were sufficiently unique spectrally to maintain their own respective class. The 11 remaining classes were categorised into two new classes, with certain forage types (i.e. AL, OG and WS = alfalfa, orchard grass and wheat straw, respectively) falling in both new classes. Using the new six-class structure (but with the same pre-treatment) to classify these samples by a linear discriminant analysis model of PCs 1–10 scores, we found that accuracy improved from 96.6% and 95.5% (cross-validation and test sets, respectively) to 98.5% and 99.6% with respect to the original 14 classes.

For a final test, discriminant analysis using the six groupings of the forage samples described above was carried out using PLS under GRAMS and SAS® using the same spectral data, classification data and method of class encodings (for example, 1,0,0,0,0,0 etc.). Examination of the results of this direct comparison for both calibration and validation/test sets using the six-class structure showed the SAS® results to be essentially identical to those produced using GRAMS software[13] with predicted and actual values for classes to be the same to three or more decimal places.

Table 3. Summary of information printed out by SAS showing settings for a specific program run.

Variable	Example Value	Description
DIRECT	C:\SASPLSPCR	Directory on computer for input and output
FILE1	ODTRT1	Name of first files in dataset
INDEX	700	Total number of data files
BEGIN	1	First data point in spectral file to use in calibrations
ENDAT	700	Last data point in spectral file to use in calibrations
AVER	4	Number of spectral data points to average or skip
NUMOFFILES	241	Total number of data files
NUMOFVARS	6	Total number of analytes in data set
FIRSTVAR	1	First analyte to use for discrimination
LASTVAR	6	Last analyte to use for discrimination
PLOTSTART	2500	First wavelength in spectral to plot
PLOTEND	1100	Last wavelength in spectra to plot
PLOTFACTOR	100	Defines X axis values for spectral plots, every 100 etc.
NUMOFCALFILES	161	How many files are to be used for calibration development
NUMOFVALFILES	80	How many files are to be used as a separate validation or test set
STARTVALFILES	162	First file used as a validation file
CV	ONE	Method of cross validation
METHOD	PLS	Discriminant algorithm used: PLS, PCR or RRR
FILEPRE	ALL241FEEDSIX	File name used for input files without the extension
FILERUN	FORPAPERBYSTEVE	Suffix used for output files for a specific run setup
RANDOMQQ	25	Specific iteration for multiples runs
RANDOMIZE	YES OR NO	Whether sample set was randomised prior to processing.
START1STGAP	2	One-half smallest gap used for gap first derivatives
END1STGAP	64	Largest gap to use for gap first derivative
COUNT1ST	5	Number of first gap first derivatives tried
START2NDGAP,EN2NDGAP,COUNT2ND	2,64,5	Same as for first derivative but for gap second derivatives
ODDEVEN	EVEN	Derivatives computed used odd or even data points to start
SAVISTYG	NON	Were Savitzky–Golay derivatives performed
MULTI	YES	Was multiplicative scatter correction performed?
STAND	NON	Was standard normal variate treatment performed?
DETRENT	NON	Was detrend performed?
MEANCENTER	SAS	How was mean centred and variance scaling performed
FIRSTSGTABLE,LASTSGTABLE,BYTABLE,STARTSG,ENDSG	1,10,1,5,25	Settings for Savitzky–Golay derivatives.

Finally, for PLS classification, in order to evaluate the results better, a macro ("CLASSSUMMARY") was written that tabulated the number of misclassified samples as either NEGATIVES (actual value = 1, predicted < .55) or FALSEPOS (actual value = 0, predicted ≥ .45). The values of .55 and .45 were not selected on any statistical basis and can be changed by the user as desired.

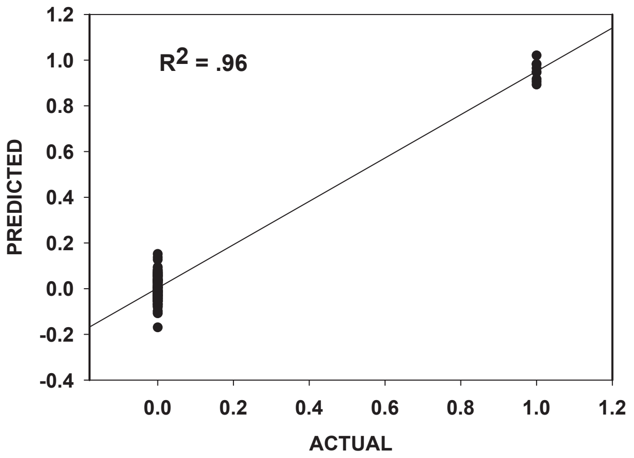

Figure 1. Selected typical predicted versus actual plot.

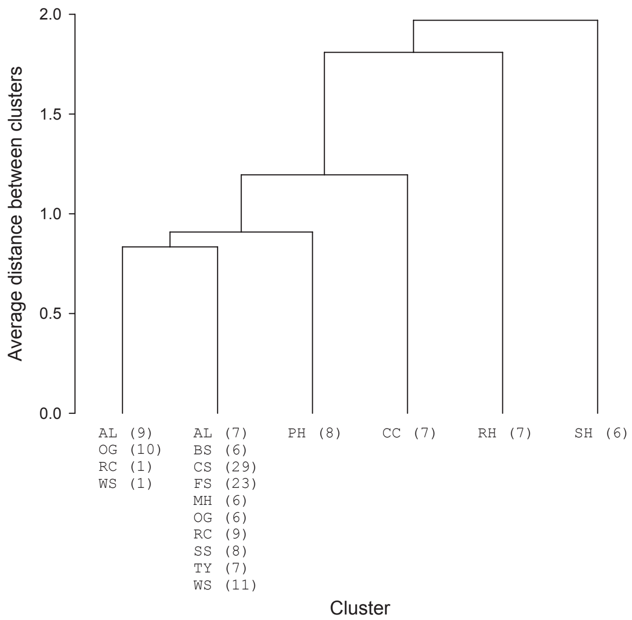

Figure 2. Dendrogram of scores 1–10 of PCA of 161 forage sample spectra initially treated with an 11-point quadratic Savitzky–Golay second derivative convolution function. Beneath each of the six clusters is a list of the forage types and the number of samples of each (AL = alfalfa hay, BS = Barley straw, CC = corn cobs, CS = corn stover, FS = Fescue, MH = Mixed hay, OG = orchard grass, PH = peanut hulls, RC = Red Clover hay, RH = rice hulls, SH = soybean hulls, SS = Soybean stover, TY = timothy, WS = wheat straw).

Conclusions

The objective of this work was to implement discriminant analysis using SAS® PLS regression for analysis of spectral data. This was done in conjunction with previous research that implemented data pre-treatments including scatter correction, derivatives, mean centring and variance scaling for spectral analysis. The program allows for testing of multiple spectral pre-treatments in a one-step fashion with summary of all results. Finally, the macro coding for the program and test data sets are available at: http://www.impublications.com/nir/page/software. While not a component of this study, PCR and reduced rank regression (RRR) can easily be implemented using the same SAS® program as described in this work.

References

1. P.J. Gemperline, L.D. Webber and F.O. Cox, "Raw materials testing using soft independent modeling of class analogy analysis of near-infrared reflectance spectra", *Anal. Chem.* **61**, 136–144 (1989). doi: 10.1021/ac00177a012
2. M. Barker and W. Rayens, "Partial least squares for discrimination", *J. Chemometr.* **17**, 166–173 (2003). doi: 10.1002/cem.785
3. J.B. Reeves, III and S.R. Delwiche, "SAS partial least squares regression for analysis of spectroscopic data", *J. Near Infrared Spectrosc.* **11**, 415–431 (2003).
4. J.B. Reeves, III and S.R. Delwiche, "Using SAS PLS calibrations for spectroscopic data", *NIR news* **15(3)**, 10–13 (2004).
5. J.J. Meulman, *Optimal scaling methods for multivariate categorical analysis.* http://www.spss.com/downloads/Papers.cfm?List=all&Name=all
6. P. Geladi, D. MacDougall and H. Martens, "Linearisation and scatter correction for near-infrared reflectance spectra of meat", *Appl. Spectrosc.* **39**, 491–500 (1985). doi: 10.1366/0003702854248656
7. R.J. Barnes, M.S. Dhanoa and S.J. Lister, "Standard normal variate transformations and de-trending of near-infrared diffuse reflectance spectra", *Appl. Spectrosc.* **43**, 772–777 (1989). doi: 10.1366/0003702894202201
8. A. Savitzky and M.J.E. Golay, "Smoothing and differentiation of data by simplified least squares procedures", *Anal. Chem.* **36**, 1627–1639 (1964).
9. J.B. Reeves, III, "Near infrared reflectance spectroscopic analysis of sodium chlorite treated forages and other plant materials", *J. Dairy Sci.* **71**, 143–151 (1988).
10. J.B. Reeves, III, "Near infrared spectroscopic analysis of lignin components in sodium chlorite treated and non-treated forages and forage by-products", *J. Dairy Sci.* **71**, 388–397 (1988).
11. S.R. Delwiche, R.A. Graybosch, L.E. Hansen, E. Souza and F.E. Dowell, "Single kernel near-infrared analysis of tetraploid (durum) wheat for classification of the waxy condition", *Cereal Chem.* **83**, 287–292 (2006). doi: 10.1094/CC-83-0287
12. *SAS User's Guide Ver. 9.1.3*, Cross validation. pp. 3384–3386 (2006) at http://support.sas.com/documentation/onlinedoc/91pdf/sasdoc_91/stat_ug_7313.pdf
13. PLSplus/IQ for GRAMS/32 and GRAMS386. Galactic Industries Corp., Salem, NH, USA, p. 73 (1996).
14. H. van der Voet, "Comparing the predictive accuracy of models using a simple randomization test", *Chemometr. Intell. Lab. Syst.* **25**, 313 (1994). doi: 10.1016/0169-7439(94)85050-X

Appendix

Existing framework

Macros from previous studies that are used in the present study are as follows: GETCHEM (reads file containing the analyte values), OVERHEAD (prepares analyte and spectral values for processing), GAPSDERIVATIVES (performs Savitzky–Golay and/or gap derivatives, SKIPAVER (averages or skips spectral data points), MSCSNV (performs multiplicative scatter correction and/or standard normal variate (SNV) correction with or without detrend or detrend without (SNV), MEANCENTERVARSCALED (determines whether mean centring and variance scaling is performed before PLS, not at all or during each iteration of the PLS algorithm), PREPAREFORPLS (prepares all data for PLS processing). Code for these previous programs and the newer discriminant version is available at the website: http://www.impublications.com/nir/page/software.

New framework

There are now four macros provided for importing spectral data into the SAS® program, as opposed to one in the original program.[3] Macros GETSPECONE and GETSPECTWO import new-type and old-type GRAMS (Galactic Industries, Salem, NH, USA) spc multifiles.[13] More than one type of spc multifile format exists and these two macros cover two forms. It should be noted that the authors have been told there may be even more forms that the two handled here due to slight variations in how different instruments are writing the files. GETSPECTHREE imports spectral files in the form of a Foss-NIRSystems NSAS/Vision file (Foss-NIRSystems, Laurel, MD, USA). Finally, macros ALTERNATEINPUT and ALTERNATIINPUTII are presented to input data from a permanent SAS® file containing the grain data. In this case, the spectral and analyte data were in the same file and the GETCHEM macro was not needed. For GETSPEC: ONE, TWO and THREE, the only requirements are that the spectral data exist in the format specified and be in the same order as the analyte data accessed by the macro GETCHEM; see earlier work for details.[3,4] If spectral data are in any formats other than those covered by the three GETSPEC macros, the user must write a routine, such as the macro ALTERNATEINPUT used here, for accessing the spectral data. In order for the spectral data to be properly accessed, they must be readable by the code below as presented in the last lines of the macro ALTERNATEINPUT:

DATA WORKFILE; INFILE 'FILENAME';
INPUT &VARLIST;

Where the variables in the macro variable &VARLIST are as follows:

ID $ IDLABEL $ &CHEMS ABS1 – ABS&MAXINDEX.

Where

ID = Input file name.

IDLABEL = identification tag for file, i.e. which class is the sample in.

&CHEMS = class coding for the sample in question, i.e. 1,0,0,0 etc.

ABS1 – ABS&MAXINDEX = spectral data values

While not a requirement, for this data set, the chemical data and spectral data were in the same file and thus read in as one set. If any of the three SPEC macros were used, the class coding would be in a separate file read by the GETCHEM macro with the data in the following format:

Filename1 IDLABEL class1code class2code class3code class-4code…
Filename2 IDLABEL class1code class2code class3code class-4code…
Filename3 IDLABEL class1code class2code class3code class-4code…
etc.

Notes

1. All the input data do not have to be on the same line, but the order of the variables is fixed and must be as outlined above.
2. As presently implemented, there can be several sets of class codes in the same dataset from which the macro variables FIRSTCLASS and LASTCLASS (lines 17 and 18 in the program) then select the specific ones for processing by PLS, e.g. have 16 analytes coded 0 or 1, but 1–4 = type of forage, 5–16 = type of processing etc. Discriminant analysis would then be carried out with FIRSTCLASS = 1 or 5 and LASTCLASS = 4 or 16, respectively, with each setup a separate run of the SAS® program.
3. In order to process data, the following files must be in the same subdirectory identified on line 20 of the program: note that calls to code in (b) and (c) will not work in Unix systems as the program is written, also RECODPLS.EXE would likely need to be recompiled using a Unix-based PASCAL compiler.
   (a) SAS® DISCRIM program
   (b) SAVGOLGAP.PRN (tables for Savitzky–Golay derivatives)
   (c) RECODPLS.EXE (sets up PLS once the number of factors to use has been determined)
   (d) Spectral and analyte files
   (e) A subdirectory named "GRAPHS" if plots are created
   This program, as with the original program,[3] operates in steps. 1. The spectral and analyte data are read into SAS® and combined into one file, or are read in together in one file. 2. The "OVERHEAD" macro is run, which creates a structured file ("WORKFILE") that is used by all subsequent macros. 3. The spectra pretreatments (MSC etc.) desired are executed and the resulting data are added to the "WORKFILE". The resulting file contains the original spectral data and a copy of each pre-treated spectral set as defined by the pre-treatment macros implemented. This final file can be stored as a permanent SAS® dataset if desired for future access, although the time needed to perform all the pre-treatments even when large numbers of pre-treatments are implemented is small compared to the time required to execute the PLS, especially when doing discriminant analysis. 4. PLS, PCA or RRR is run and the optimal number of factors is determined using two different criteria. The first is a method used by SAS®.[12] This

selection is based a significance test suggested by van der Voet for determining whether models with additional factors are significantly different than those with fewer.[14] The second is based on the F-test method used in GRAMS.[13] In general, the number of factors determined by the SAS® method has been found to generally be either the same or one greater than the number determined by the F-test method. Once the number of factors to be used is determined, PLS is run again using the selected number of factors. The file program RECODPLS.EXE creates the SAS® code for rerunning the PLS with the optimal number of factors. *This program is a compiled PASCAL program and as noted is compiled for DOS, not Unix.* As described in the original PLS manuscript,[3] the code in this program may need to be changed if options other than those defined in program lines 56–57 are to be implemented (see program documentation file for further details). For those without a Pascal compiler, a free compiler (used for this program) can be downloaded from http://www.bloodshed.net/devpascal.html.

Processor time

At the 2006 International Diffuse Reflectance Conference in Chambersburg, PA, USA, a soil set containing 2761 calibration samples was used to examine the different ways in which people carry out calibrations in a session called the "Software Shootout" (http://www.idrc-chambersburg.org/shootout.htm). Some analysts found that efforts to run PLS regressions appeared to freeze their computers using several available commercial chemometrics packages so the time taken by SAS® for the grain dataset does not appear to be unusual. (We say "appeared" because people gave up when nothing happened for hours or days.) The same effect has been seen with this program, as there is often no evidence that anything is occurring while the program executes.